The Accommodating Function
- A Generalization of the Competitive Ratio

Joan Boyar*, Kim S. Larsen*, and Morten N. Nielsen

University of Southern Denmark, Odense, Denmark,**
{joan,kslarsen,nyhave}@imada.sdu.dk

Abstract. A new measure, the accommodating function, for the quality of on-line algorithms is presented. The accommodating function, which is a generalization of both the competitive ratio and the accommodating ratio, measures the quality of an on-line algorithm as a function of the resources that would be sufficient for some algorithm to fully grant all requests. More precisely, if we have some amount of resources n, the function value at α is the usual ratio (still on some fixed amount of resources n), except that input sequences are restricted to those where all requests could have been fully granted by some algorithm if it had had the amount of resources αn. The accommodating functions for two specific on-line problems are investigated: a variant of bin-packing in which the goal is to maximize the number of objects put in n bins and the seat reservation problem.

1 Introduction

The competitive ratio [11, 19, 15], as a measure for the quality of on-line algorithms, has been criticized for giving bounds that are unrealistically pessimistic [1, 2, 12, 14, 16], and for not being able to distinguish between algorithms with very different behavior in practical applications [2, 12, 16, 18]. Though this criticism also applies to standard worst-case analysis, it is often more disturbing in the on-line scenario [12].

The basic problem is that the adversary is too powerful compared with the on-line algorithm. For instance, it would often be more interesting to compare an on-line algorithm to other on-line alternatives than to an all-powerful off-line algorithm. A number of papers have addressed this problem [10, 2, 6, 13, 14, 18, 11, 21, 20, 16] by making the on-line algorithm more powerful, by providing the on-line algorithm with more information, or by restricting input sequences.

In this paper, we move in the direction of restricting input sequences. However, instead of a "fixed" restriction, we consider a function of the restriction, the accommodating function. Informally, in on-line problems, where requests are

* Supported in part by the ESPRIT Long Term Research Programme of the EU under project number 20244 (ALCOM-IT) and in part by SNF, Denmark.

** Dept. of Mathematics and Computer Science, University of Southern Denmark, Main campus: Odense University, Campusvej 55, DK–5230 Odense M, Denmark.
made for parts of some resource, we measure the quality of an on-line algorithm as a function of the resources that would be sufficient for an optimal off-line algorithm to fully grant all requests. More precisely, if we have some amount of resources \( n \), the function value at \( \alpha \) is the usual ratio (still on some fixed amount of resources \( n \)), except that input sequences are restricted to those where all requests could have been fully granted by an optimal off-line algorithm if it had had the amount of resources \( \alpha n \).

In the limit, as \( \alpha \) tends towards infinity, there is no restriction on the input sequence, so this is the competitive ratio. When all requests can be fully granted by an optimal off-line algorithm (without using extra resources), the function value is the accommodating ratio \([3]\). Consequently, the accommodating function is a true generalization of the competitive as well as the accommodating ratio.

In addition to giving rise to new interesting algorithmic and analytical problems, which we have only begun investigating, this function, compared to just one ratio, contains more information about the on-line algorithms. For some problems, this information gives a more realistic impression of the algorithm than the competitive ratio does. Additionally, this information can be exploited in new ways. The shape of the function, for instance, can be used to warn against critical scenarios, where the performance of the on-line algorithm compared to the off-line can suddenly drop rapidly when the number of requests increases.

Due to space limitations, all proofs have been eliminated from this paper. They can be found in \([4]\).

2 The Accommodating Function

Consider an on-line problem with a fixed amount of resources \( n \). Let \( \rho(I) \) denote the minimum resources necessary for an optimal off-line algorithm to fully grant all requests from the request sequence \( I \). We refer to \( I \) as an \( \alpha \)-sequence, if \( \rho(I) \leq \alpha \cdot n \), i.e., an \( \alpha \)-sequence is a sequence for which an optimal off-line algorithm could have fully granted all requests, if it had had resources \( \alpha \cdot n \).

For a maximization problem, \( A(I) \) is the value of running the on-line algorithm \( A \) on \( I \), and OPT(I) is the maximum value that can be achieved on \( I \) by an optimal off-line algorithm, OPT. Note that \( A \) and \( OPT \) use the same amount of resources, \( n \).

The algorithm \( A \) is \( c \)-accommodating w.r.t. \( \alpha \)-sequences if \( c \leq 1 \) and for every \( \alpha \)-sequence \( I \), \( A(I) \geq c \cdot\) OPT(I) \(- b \), where \( b \) is a fixed constant for the given problem, and, thus, independent of \( I \).

Let \( C_\alpha = \{ c \mid A \) is \( c \)-accommodating w.r.t. \( \alpha \)-sequences\}. The accommodating function \( A \) is defined as \( A(\alpha) = \sup C_\alpha \).

For a minimization problem, \( A(I) \) is a cost and OPT(I) is the minimum cost which can be achieved. Furthermore, \( A \) is \( c \)-accommodating w.r.t. \( \alpha \)-sequences if \( c \geq 1 \) and for every \( \alpha \)-sequence \( I \), \( A(I) \leq c \cdot\) OPT(I) \(+ b \), and the accommodating function is defined as \( A(\alpha) = \inf C_\alpha \).